

	🔖 Вид	⌵ Время жизни	⌵ Описание	⌵ Инвалидация	+
	Request Memoization	Кеш действует в рамках одного запроса	переиспользует результат fetch из разных компонентов/ generateMetadata. После запрос кеш сбрасывается. Подходит только для GET запросов, если у вас POST (например, graphql), то нужно оборачивать в react cache	-	
	Data cache	Кеш действует между запросами и между релизами	к запросу нужно добавить cache: "force-cache"	инвалидировать можно только с помощью по времени (revalidate) или по запросу (revalidatePath или revalidateTag)	
	Full route cache	Кешируется между запросами	Хранение результатов рендера RSC payload (на сервере). Кешируются только статичные сегменты.	Ивалидируется после редеплоя или инвалидации Data Cache	
	Client-side router cache	Действует только в рамках сессии (при обновлении страницы сбрасывается)	Хранит RSC payload на клиенте. Layout, loading states по умолчанию переиспользуются, а страницы перерисовываются (но при навигации вперед/назад по стрелкам не перерисовываются + есть настройка staleTimes)	Инвалидация - автоматическая в зависимости от параметры prefetch  prefetch={null} or unspecified: not cached for dynamic pages, 5 minutes for static pages.  prefetch={true} or router: prefetch: 5 minutes for both static & dynamic pages. И ручная: в server actions revalidatePath или revalidateByTag, cookies.set или cookies.delete	
+					

